

## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

УДК 004.7:004.032.26

В. М. ПАХОМОВА<sup>1\*</sup>, І. Д. ЦИКАЛО<sup>2\*</sup>

<sup>1\*</sup>Каф. «Електронні обчислювальні машини», Дніпропетровський національний університет залізничного транспорту імені академіка В. Лазаряна, вул. Лазаряна, 2, Дніпро, Україна, 49010, тел. +38 (056) 373 15 89, ел. пошта viknikpakh@gmail.com, ORCID 0000-0002-0022-099X

<sup>2\*</sup>Каф. «Електронні обчислювальні машини», Дніпропетровський національний університет залізничного транспорту імені академіка В. Лазаряна, вул. Лазаряна, 2, Дніпро, Україна, 49010, тел. +38 (056) 373 15 89, ел. пошта ihor.tsykalo@gmail.com, ORCID 0000-0002-1629-5873

### ВИЗНАЧЕННЯ ОПТИМАЛЬНОГО МАРШРУТУ В КОМП'ЮТЕРНІЙ МЕРЕЖІ ЗАСОБАМИ БАГАТОШАРОВОЇ НЕЙРОННОЇ МОДЕЛІ

**Мета.** Класичні алгоритми пошуку найкоротшого шляху на графі, що лежать в основі наявних протоколів маршрутизації, які сьогодні використовують у комп'ютерних мережах, в умовах постійної зміни завантаженості мережі не можуть привести до оптимального рішення в реальному часі. У зв'язку з цим метою статті є розробити методику визначення оптимального маршруту в об'єднаній комп'ютерній мережі. **Методика.** Для визначення оптимального маршруту в об'єднаній комп'ютерній мережі, що працює за різними технологіями, розроблено на мові Python із використанням фреймворку TensorFlow програмну модель «MLP 34-2-410-34». Вона дозволяє виконувати наступні етапи: генерацію вибірки (випадкову або збалансовану); створення нейронної мережі, на вхід якої подають масив пропускних спроможностей каналів комп'ютерної мережі; навчання й тестування нейронної мережі на відповідних вибірках. **Результати.** Нейронна мережа конфігурації 34-2-410-34 з функціями активації ReLU та Leaky-ReLU у прихованому шарі та лінійною функцією активації у вихідному шарі навчається за алгоритмом Adam. Цей алгоритм є комбінацією алгоритмів Adagrad, RMSprop та стохастичного градієнтного спуску з інерцією. Зазначені функції навчаються найбільш швидко на всіх обсягах навчальної вибірки, менш за інші піддаються перенавчанню, й досягають значення помилки в 0,0024 на контрольній вибірці й у 86 % визначає оптимальний шлях. **Наукова новизна.** Проведено дослідження параметрів нейронної мережі на основі розрахунку середнього гармонійного за різних функцій активації (Linear, Sigmoid, Tanh, Softplus, ReLU, L-ReLU) на навчальних вибірках різного обсягу (140, 1 400, 14 000, 49 000 прикладів) та за різними алгоритмами оптимізації навчання нейронної мережі (BGD, MB SGD, Adam, Adamax, Nadam). **Практична значимість.** Використання нейронної моделі, на вхід якої подають значення пропускних спроможностей каналів, дозволить у реальному часі визначити оптимальний маршрут в об'єднаній комп'ютерній мережі.

**Ключові слова:** комп'ютерна мережа; оптимальний маршрут; нейронна мережа; вибірка; середнє гармонійне; функція активації; алгоритм оптимізації

#### Вступ

Одною з основних вимог, які висувають до алгоритмів маршрутизації, є їх швидка збігальність до оптимального рішення, що продиктовано необхідністю їх протокольної реалізації

в реальному масштабі часу в умовах постійної зміни характеристик мережного трафіка, топології й завантаженості комп'ютерних мереж, які використовують на залізничному транспорті. Класичні алгоритми пошуку найкоротшого шляху на графі, що використовуються в сучасних протоколах маршрутизації, не можуть цю-

## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

го зробити. Одним із підходів розв'язання задач маршрутизації в комп'ютерних мережах є використання нейромережної технології [8, 15–16]. Так, наприклад, у [12] показано, що за допомогою нейронної мережі (НМ) можна забезпечити знаходження близького до оптимального розв'язання задачі комівояжера та пошуку найкоротшого шляху на графі. У [3] досліджена можливість використання для розв'язання задач маршрутизації наступних нейронних мереж: багатосарового перцептрона (Multi Layer Perceptron); мережі RBF; мережі Хопфілда. Установлено, що найбільш перспективними засобами розв'язання задачі маршрутизації є нейронна мережа прямого розповсюдження та мережа Хопфілда, які здатні працювати в умовах динамічної зміни топології комп'ютерної мережі та характеристик каналів передачі даних [1–2]. Зокрема, під час використання мережі Хопфілда необхідно провести додаткові дослідження передавальних функцій нейронів і функції енергії нейронної мережі [18]. У роботі [7] виявлено, що мережа Хопфілда знаходить задовільний маршрут, який відрізняється від оптимального в середньому на 7–8 % (у випадку кількості місць більше ніж 15). Проаналізована можливість використання мережі Хопфілда для пошуку найкоротшого шляху на графі маршрутів у комп'ютерній мережі залізничного транспорту [5–6]. У [3] досліджено використання нейронної мережі прямого розповсюдження, що створена в MatLAB, із метою визначення маршруту в комп'ютерній мережі з п'яти вузлів. Але об'єднана комп'ютерна мережа залізничного транспорту складається зі значно більшої кількості вузлів, що потребує проведення додаткових досліджень. Зокрема, в [20] запропонована інтелектуальна підсистема управління з використанням мережної технології, в [17] – підсистема прогнозування на основі нейронечіткої мережі.

**Мета**

Розробити методіку визначення оптимального маршруту в об'єднаній комп'ютерній мережі на основі створеної програмної моделі «MLP34-2-410-34» за допомогою фреймворку TensorFlow.

**Методика**

Об'єднану комп'ютерну мережу, що працює за різними технологіями, можна представити у вигляді неорієнтованого графа  $G(V, W)$ , де  $V$  – множина вершин графа, кількість яких  $N$ , причому кожна вершина моделює собою вузол (маршрутизатор) комп'ютерної мережі;  $W$  – множина ребер графа, кількість яких  $M$ . Кожному ребру графа присвоюється певна вага, яка відповідає пропускній спроможності (максимальний обсяг даних, переданий мережею за одиницю часу):

$$C = \{c_{ij}\}, \quad (1)$$

де  $c_{ij}$  – пропускна спроможність каналу зв'язку між  $i$ -ми та  $j$ -ми вузлами мережі, Мбіт/с.

Для розв'язання задачі маршрутизації потрібно знайти оптимальний шлях між двома заданими маршрутизаторами об'єднаної комп'ютерної мережі. Як приклад розглянемо гіпотетичну комп'ютерну мережу, структура якої наведена на рис. 1.

Уведемо масив:

$$X = \{x_{ij}\}, \quad (2)$$

де  $x_{ij}$  – наявність трафіка, що передається в мережу між  $i$ -ю та  $j$ -ю вершинами. Як обмеження  $x_{ij} \in \{0,1\}$ , тобто змінна приймає значення 1, якщо трафік протікає по каналу  $(i, j)$ ; в іншому випадку – 0.

Як критерій виступає наступний вираз:

$$\min_x F \cdot X, \quad (3)$$

де  $f_{ij} = \frac{125\,000\,000}{C_{ij}}$ , що гарантує пошук шляху з максимальною пропускною спроможністю.

Якщо між вузлами об'єднаної комп'ютерної мережі відсутній зв'язок, то  $c_{ij} = c_{ji} = 0$  (таким чином,  $f_{ij} = \infty$ ).

## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

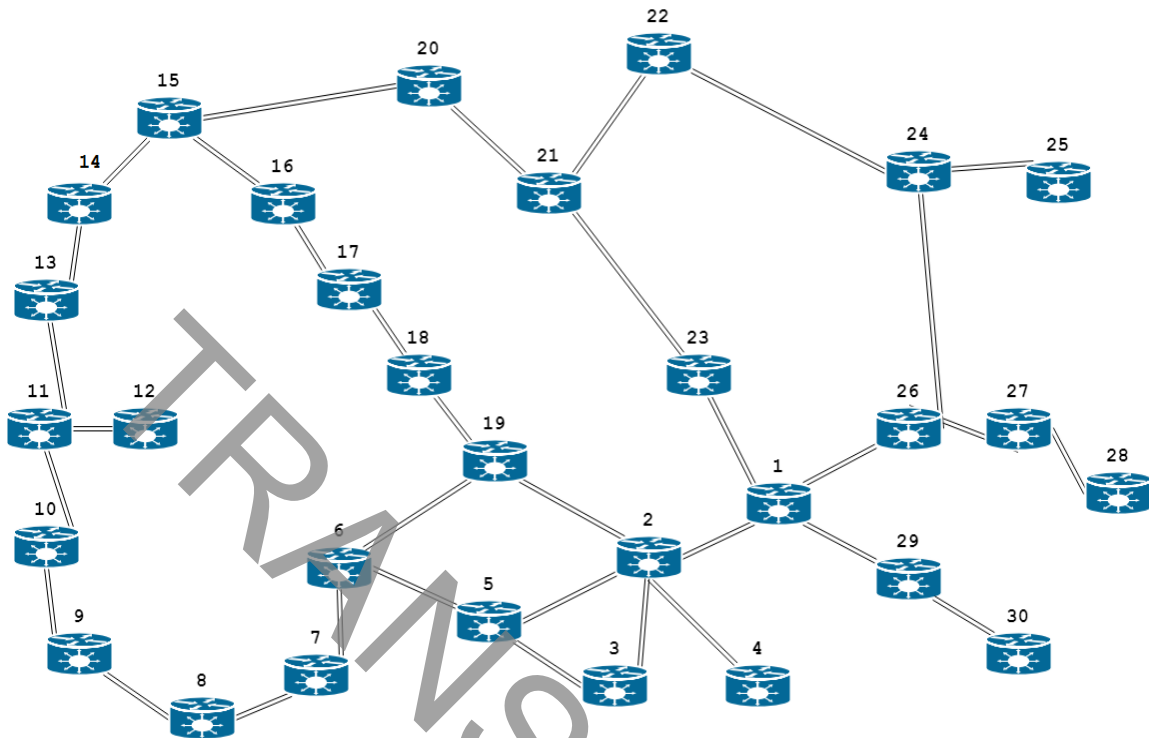


Рис. 1. Граф з'єднань маршрутизаторів об'єднаної комп'ютерної мережі

**Результати**

Нейронна мережа як основний математичний апарат розв'язання задачі. В об'єднаній комп'ютерній мережі нараховується 30 маршрутизаторів та 34 канали зв'язку. Як приклад розглянемо розв'язання задачі визначення оптимального маршруту між вузлами «12» та «1». Узагалі між зазначеними вузлами існує 14 унікальних шляхів.

- Шлях 1: [12, 11, 13, 14, 15, 20, 21, 23, 1];
- шлях 2: [12, 11, 10, 9, 8, 7, 6, 19, 2, 1];
- шлях 3: [12, 11, 10, 9, 8, 7, 6, 5, 2, 1];
- шлях 4: [12, 11, 10, 9, 8, 7, 6, 5, 3, 2, 1];
- шлях 5: [12, 11, 13, 14, 15, 16, 17, 18, 19, 2, 1];
- шлях 6: [12, 11, 13, 14, 15, 20, 21, 22, 24, 26, 1];
- шлях 7: [12, 11, 13, 14, 15, 16, 17, 18, 19, 6, 5, 2, 1];
- шлях 8: [12, 11, 13, 14, 15, 16, 17, 18, 19, 6, 5, 3, 2, 1];
- шлях 9: [12, 11, 10, 9, 8, 7, 6, 19, 18, 17, 16, 15, 20, 21, 23, 1];
- шлях 10: [12, 11, 10, 9, 8, 7, 6, 19, 18, 17, 16, 15, 20, 21, 22, 24, 26, 1];
- шлях 11: [12, 11, 10, 9, 8, 7, 6, 5, 2, 19, 18, 17, 16,

15, 20, 21, 23, 1];

шлях 12: [12, 11, 10, 9, 8, 7, 6, 5, 3, 2, 19, 18, 17, 16, 15, 20, 21, 23, 1];

шлях 13: [12, 11, 10, 9, 8, 7, 6, 5, 2, 19, 18, 17, 16, 15, 20, 21, 22, 24, 26, 1];

шлях 14: [12, 11, 10, 9, 8, 7, 6, 5, 3, 2, 19, 18, 17, 16, 15, 20, 21, 22, 24, 26, 1].

Для розв'язання задачі маршрутизації використана НМ, структура якої показана на рис. 2. На вхід НМ подають вектор пропускних спроможностей каналів об'єднаної комп'ютерної мережі  $X$ , що характеризує її поточний стан  $X = \{x_i\}$ , де  $i = 1, \dots, m$  ( $m = 34$ ). Так, наприклад, для НМ під час використання навчальної вибірки у 1 400 прикладів кількість необхідних нейронів у прихованому шарі оцінюють наступним чином:

$$\frac{34 \cdot 1\,400}{(1 + \log_2 1\,400) \cdot (34 + 34)} \leq L \leq$$

$$\leq \frac{34 \cdot \left(\frac{1\,400}{34} + 1\right) \cdot (34 + 34 + 1) + 34}{34 + 34}$$

Отже,  $62 \leq L \leq 1\,456$ .

## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

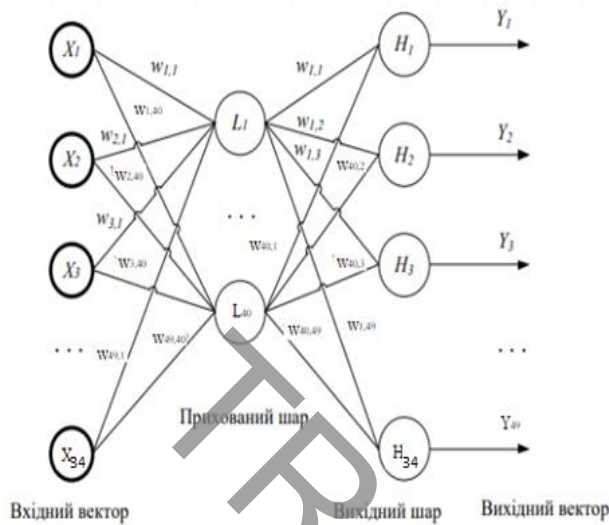


Рис. 2. Структура багатошарової НМ

*Підготовка вибірки (підготовчий етап).* Формування вибірки здійснене за фіксованої структури об'єднаної комп'ютерної мережі (див. рис. 1). Вхідний вектор  $X$  побудований шляхом генерації випадковим чином значень пропускової спроможності каналів  $c_{ij}$ , при цьому ці значення сформовані за рівномірним розподілом на сегменті [100; 100 000 000]. Вектор відгуку  $Y$  сформований шляхом розрахунку оптимального шляху за алгоритмом Дейкстри із використанням мови Python бібліотеки Networkx (відкрита програмна бібліотека, яку використовують для роботи з графами та мережами).

Вибірки побудовані таким чином, що кожен із 14 унікальних шляхів присутній із однаковою частотою. Тестова вибірка (test) нараховує 700 екземплярів, контрольна вибірка (validation) – 700 екземплярів, перша навчальна вибірка (train) – 140 прикладів (по 10 прикладів на кожен шлях), друга навчальна вибірка – 1 400 прикладів (по 100 прикладів на кожен шлях), третя навчальна вибірка – 14 000 прикладів (по 1 000 прикладів на кожен шлях), четверта навчальна вибірка – 49 000 прикладів (по 3 500 прикладів на кожен шлях).

Усі дані спочатку нормалізують до діапазону від 0 до 1 за формулою:

$$x = \frac{x - \min(X)}{\max(X) - \min(X)}, x \in X \quad (4)$$

Структура результуючого вектора наступна:  $Y = [y_{1,2}, y_{1,26}, y_{1,29}, y_{1,23}, y_{2,19}, y_{2,3}, y_{2,5}, y_{2,4}, y_{3,5}, y_{5,6}, y_{6,19}, y_{6,7}, y_{7,8}, y_{8,9}, y_{9,10}, y_{10,11}, y_{11,12}, y_{11,13}, y_{13,14}, y_{14,15}, y_{15,16}, y_{15,20}, y_{16,17}, y_{17,18}, y_{18,19}, y_{20,21}, y_{21,22}, y_{21,23}, y_{22,24}, y_{24,25}, y_{24,26}, y_{26,27}, y_{27,28}, y_{29,30}]$ , де  $y_{ij} \in \{0,1\}$ , що відповідає використанню або невикористанню відповідного каналу зв'язку в маршруті.

*Обґрунтування вибору засобів моделювання.* Для розв'язання задачі маршрутизації в об'єднаній комп'ютерній мережі обрано бібліотеку Keras із використанням TensorFlow та Numpy на мові програмування Python [9–11, 13–14, 19].

Keras – відкрита нейромережна бібліотека, написана мовою Python, що здатна працювати поверх DeepLearning4, TensorFlow та Theano, спроектованих для швидких експериментів із нейронними мережами глибинного навчання.

TensorFlow – відкрита програмна бібліотека для машинного навчання. Є системою машинного навчання Google Brain другого покоління, випущеною як відкрите програмне забезпечення.

Numpy – розширення мови Python, що надає підтримку великих багатовимірних масивів і матриць разом із бібліотекою високорівневих математичних функцій для операцій із цими масивами.

Python – інтерпретована об'єктно-орієнтована мова програмування високого рівня зі суворою динамічною типізацією. Структури даних високого рівня разом із динамічною семантикою й динамічним зв'язуванням роблять її привабливою для швидкої розробки програм, а також засобом наявних існуючих компонентів. Python підтримує модулі та пакети модулів, що сприяє модульності й повторному використанню коду. Інтерпретатор Python і стандартні бібліотеки доступні як у скомпільованій, так і у вихідній формі на всіх основних платформах. Мова програмування Python підтримує кілька парадигм програмування, зокрема: об'єктно-орієнтовану; процедурну; функціональну; аспектно-орієнтовану.

*Структура програмної моделі «MLP 34-2-410-34».* Програмна модель «MLP 34-2-40-34», що створена для моделювання й дослідження, складається із головного

## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

модуля «Main» та наступних класів: «Генератор», «MLPModel», «NetworkX», «Matplotlib», «Keras.Model», «Metrics», «Tensorflow». У головному модулі «Main» передбачені завдання меню: 1 – генерація вибірки; 2 – навчання нейронної мережі; 3 – тестування нейронної мережі. «Генератор» виконує підготовку вибірки у двох режимах: випадковий (розрахунок випадкової кількості прикладів на кожний шлях,

рис. 3, а); збалансований (однакова кількість прикладів на кожний можливий шлях, рис. 3, б). Наприклад, вибірка зі 140 прикладів за збалансованого режиму потребує 5 хвилин, у той час як за випадкового режиму це займе менше однієї хвилини.

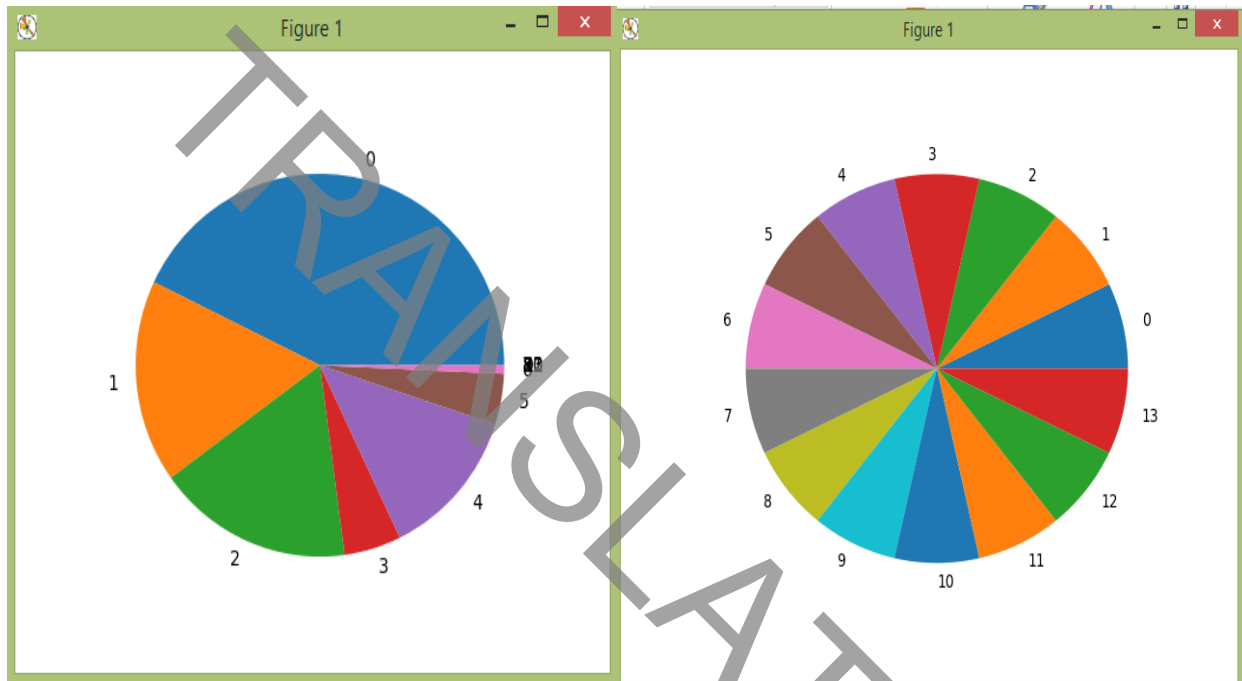


Рис. 3. Генерація вибірки:  
а – випадкової; б – збалансованої

«MLPModel» створює нейронну мережу конфігурації 34-2- $X$ -34 (де  $X$  – можлива кількість прихованих нейронів) та виконує наступні етапи: навчання; тестування; контроль на відповідних вибірках і їх нормалізація.

«NetworkX» (стандартний клас) будує граф комп'ютерної мережі (Graph), розраховує наявні шляхи між станціями (all\_simple\_paths), шлях між заданими станціями за алгоритмом Дейкстри (bidirectional\_dijkstra).

«Matplotlib» (стандартний клас) будує кругову діаграму (pie chart) та гістограму для відображення відношення кількості прикладів на кожен шлях.

«Keras.Model» (стандартний клас) виконує компіляцію відповідно до заданої конфігурації нейронної мережі (compile), представляє стандартні функції (fit, predict), які використовують під час виконання навчання й тестування нейронної мережі.

«Metrics» виконує розрахунок імовірності оптимальних та імовірності коректних відповідей.

«TensorFlow» (стандартний клас) викликається класом «Keras.Model» під час виконання відповідних розрахунків.

Загальна структура програмної моделі «MLP 34-2-40-34» подана на рис. 4.

## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

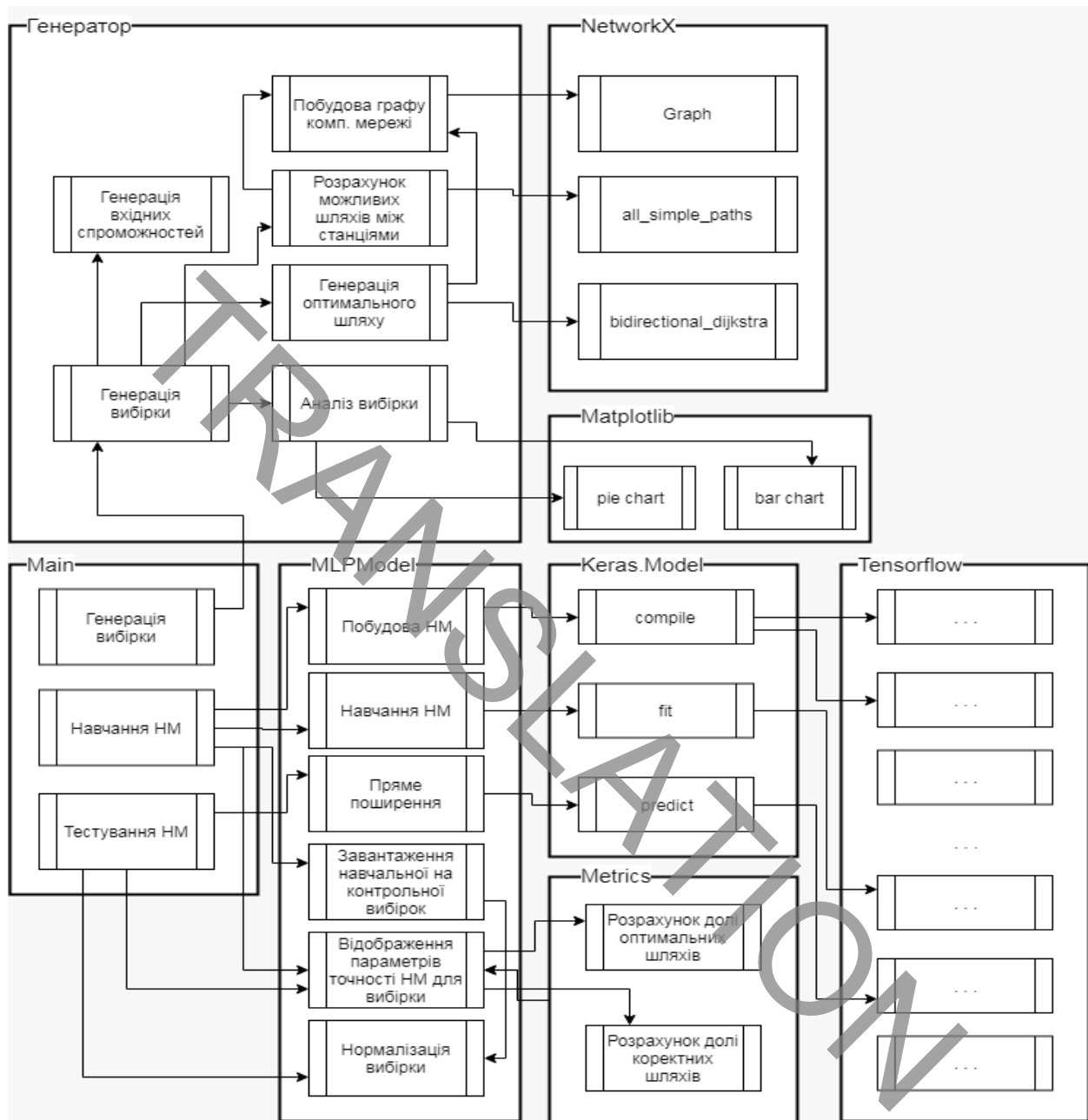


Рис. 4. Структура програмної моделі «MLP 34-2-40-34»

Для того, щоб мати можливість однозначно порівнювати моделі НМ за двома параметрами – імовірності оптимальних відповідей та імовірності коректних відповідей, – уведено значення середнього гармонійного, яке розраховують за наступною формулою:

$$H = \frac{2\rho^{opt} \rho^{kor}}{\rho^{opt} + \rho^{kor}}, \quad (5)$$

де  $\rho^{opt}$  – імовірність оптимальних відповідей,  
 $\rho^{kor}$  – імовірність коректних відповідей.

## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

Тестування програми «MLP34-2-410-34». Отриманий вектор ознак входження каналів до оптимального шляху має вигляд:  $\{0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,0,1,0,0,0,1,0,1,0,0,0,0,0\}$ , що відповідає наступним

з'єднанням маршрутизаторів у мережі: 1–23, 11–12, 11–13, 13–14, 14–15, 15–20, 20–21, 21–23. Результуючий шлях відображений на рис. 5.

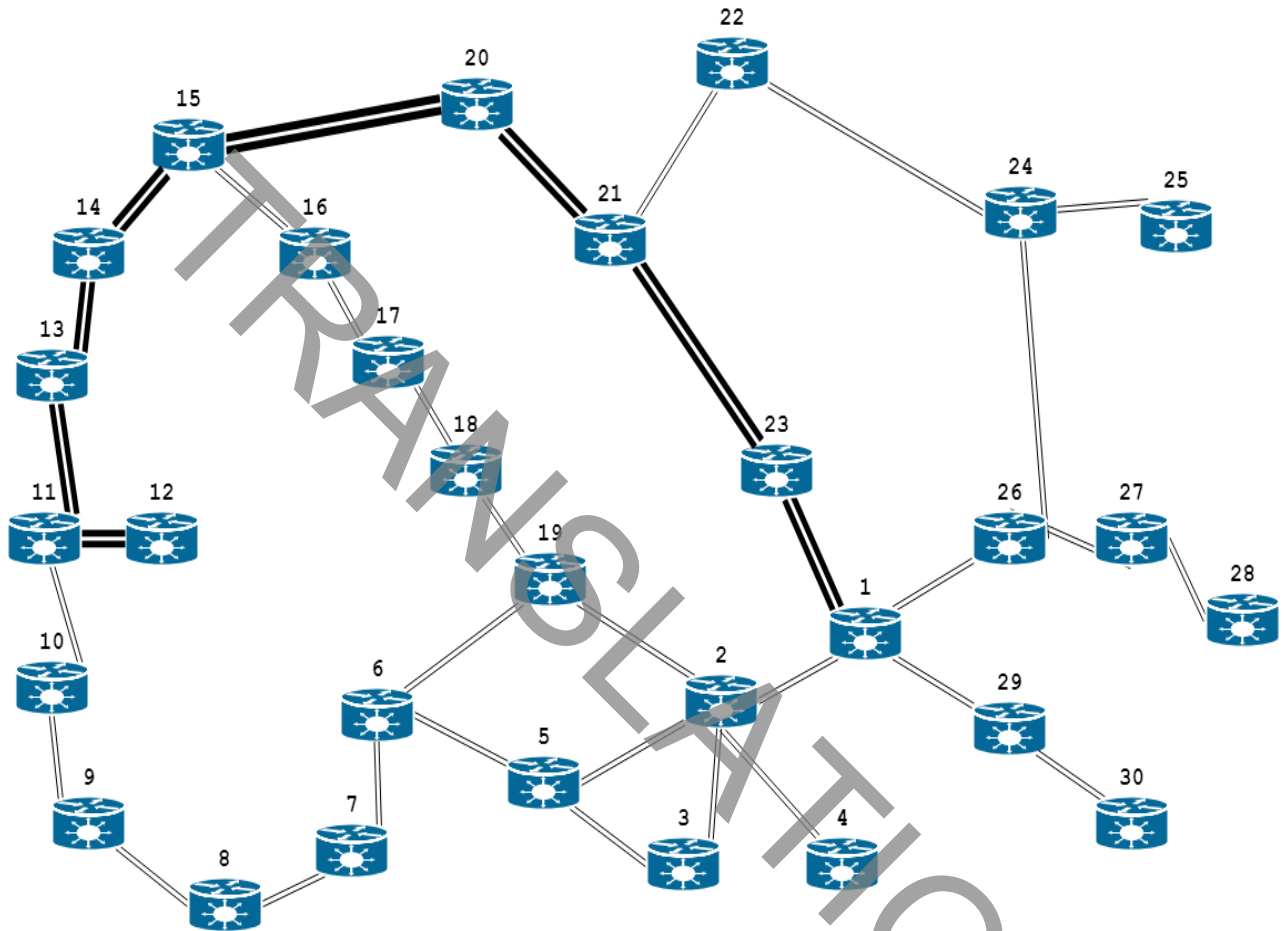


Рис. 5. Отриманий оптимальний шлях, що складається із каналів: 12–11, 11–13, 13–14, 14–15, 15–20, 20–21, 21–23, 23–1

Дослідження ефективності навчання НМ за різної кількості нейронів у прихованому шарі. Дослідження на моделі НМ проведені за алгоритмом стохастичного градієнтного спуску (розмір пакета «batchsize» 64) з оптимізацією Adam (швидкість навчання  $\alpha = 0,001$ , інерція  $\beta_1 = 0,9$ , RMSprop  $\beta_2 = 0,999$ , decay  $= 10^{-5}$ ) на 1 400 навчальних прикладах, на протязі 1 000

епох для різної кількості прихованих нейронів:  $L = \{62, 410, 760, 1110, 1456\}$ .

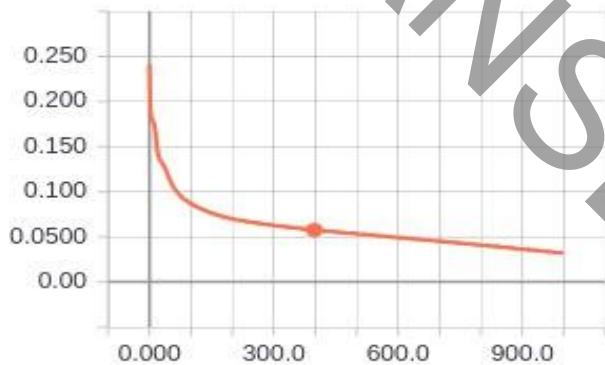
Результати експерименту наведені у табл. 1. Із таблиці видно, що для навчальної вибірки найкращий результат 0,99 досягається вже за кількості 410 нейронів у прихованому шарі.

Дослідження НМ за різної кількості прихованих нейронів

Кількість прихованих нейронів		62	410	760	1 110	1 456
Навчальна вибірка	$\rho^{opt}$	0,77	0,99	0,99	0,99	0,99
	$\rho^{кор}$	0,81	0,99	0,99	0,99	0,992
	$H$	0,79	0,99	0,99	0,99	0,99
Тестова вибірка	$\rho^{opt}$	0,55	0,52	0,51	0,52	0,51
	$\rho^{кор}$	0,71	0,7	0,69	0,7	0,71
	$H$	0,62	0,6	0,59	0,6	0,59

Процес навчання НМ проілюстровано на графіках залежності значення цільової функції від епохи для навчальної й тестової вибірок на рис. 6–10.

а



б

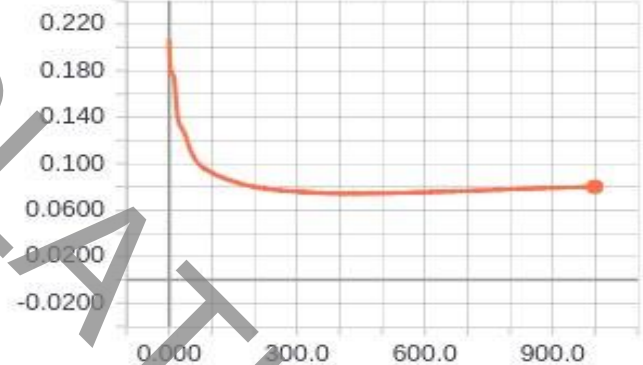
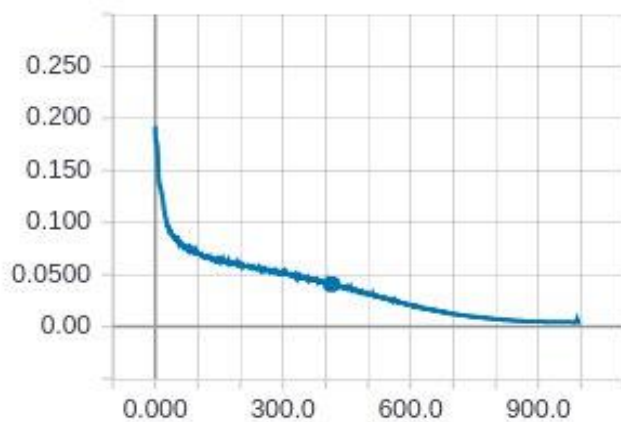


Рис. 6. Графік залежності помилки НМ 34-2-64-34 від епохи на вибірці:

а – навчальної; б – тестової

а



б

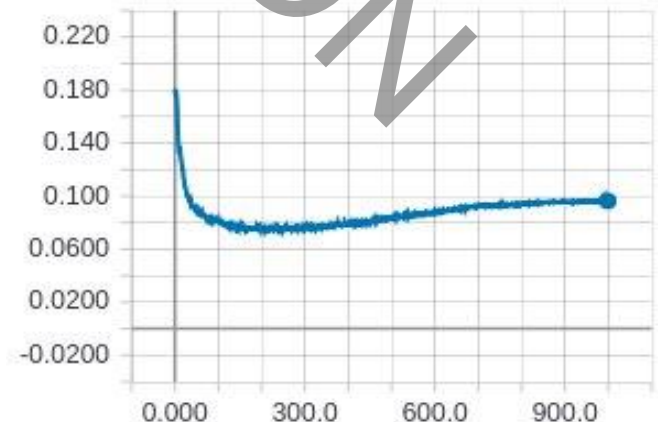


Рис.7. Графік залежності помилки НМ 34-2-410-34 від епохи на вибірці:

а – навчальної; б – тестової



## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

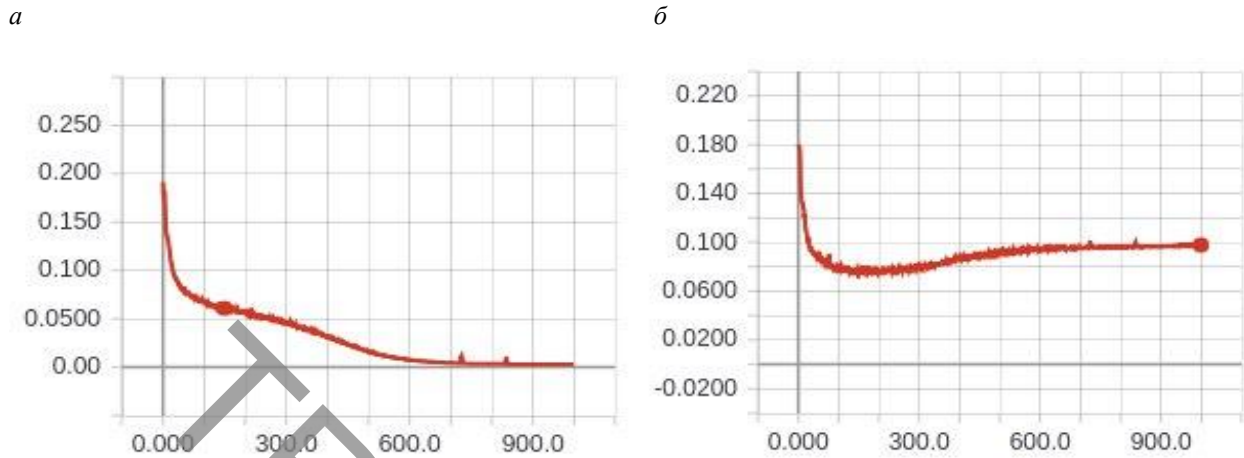


Рис. 8. Графік залежності помилки НМ 34-2-760-34 від епохи на вибірці:  
*a* – навчальної; *б* – тестової

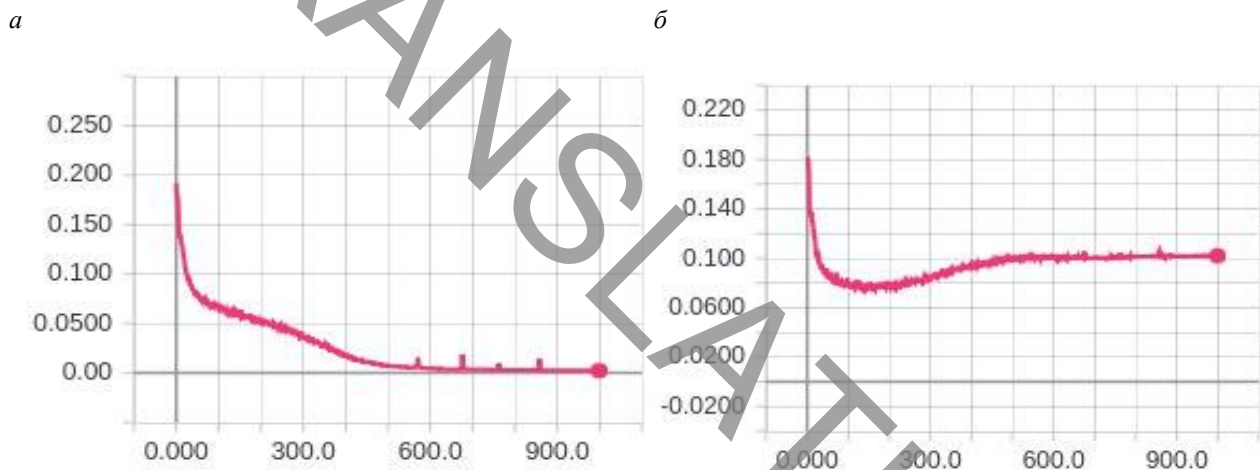


Рис. 9. Графік залежності помилки НМ 34-2-1110-34 від епохи на вибірці:  
*a* – навчальної; *б* – тестової

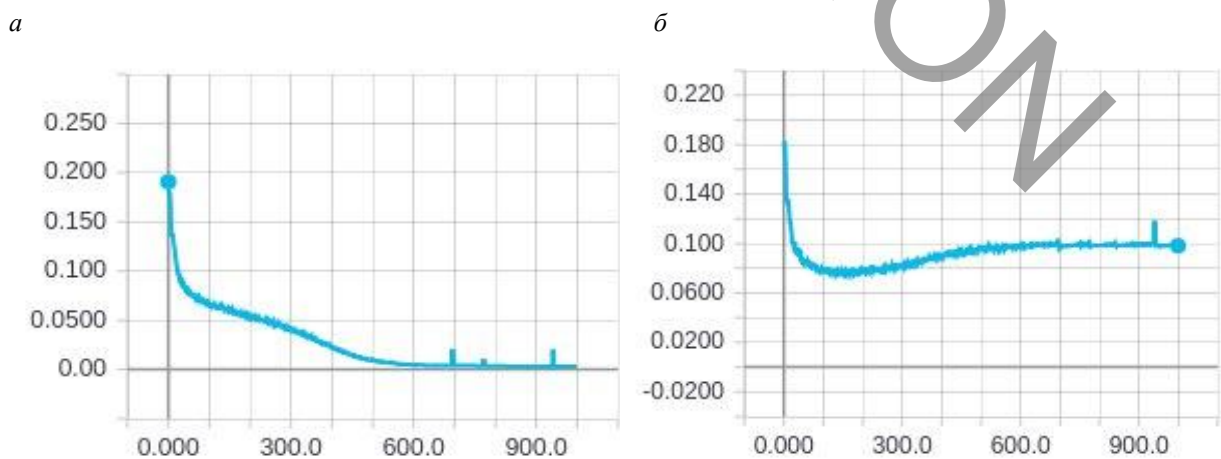


Рис. 10. Графік залежності помилки НМ 34-2-1456-34 від епохи на вибірці:  
*a* – навчальної; *б* – тестової

## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

Із рис. 6–10 видно, що для всіх запропонованих моделей НМ характерне перенавчання приблизно після 200-ої епохи. Оскільки модель НМ конфігурації 34-2-410-34 дає найкращий результат за відносно малої кількості нейронів, вона обрана для подальших досліджень як найбільш перспективна.

Дослідження ефективності навчання НМ за різних функцій активації й обсягів навчальної вибірки. Досліджено ефективність моделі НМ 34-2-410-34 за різних функцій активації нейронів у прихованому шарі. Вихідний шар має лінійну функцію активації. Навчання проведене за алгоритмом стохастичного градієнт-

ного спуску (розмір пакета «batchsize» 64) з оптимізацією Adam (швидкість навчання  $\alpha = 0,001$ , інерція  $\beta_1 = 0.9$ , RMSprop  $\beta_2 = 0.999$ ,  $\text{decay} = 10^{-5}$ ) протягом 1 000 епох. Експерименти проведені для різних функцій активації в прихованому шарі (лінійна, сигмоїдна, гіперболічний тангенс, Softplus, ReLU, Leaky-ReLU  $\alpha = 0,1$ ; табл. 2) та за різних обсягів навчальної вибірки (140, 1 400, 14 000 та 49 000 прикладів). Результати моделювання НМ наведені у табл. 2–4, процес навчання НМ проілюстровано на рис. 11–14.

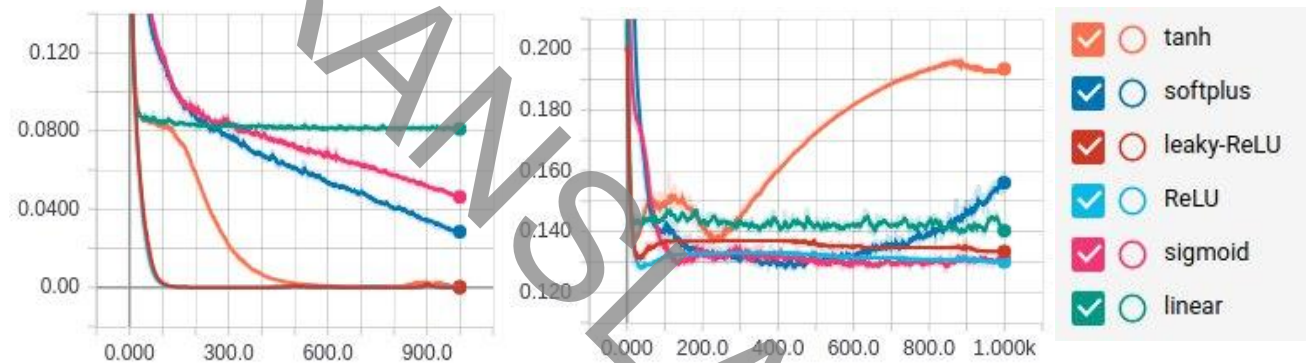


Рис. 11. Графік залежності помилки MSE від епохи для навчальної й тестової вибірок зі 140 прикладів

Дослідження показало, що навчання НМ за Softplus та Sigmoid функцій активації не припинилось, і є можливість досягнути більшої

точності під час подальшого навчання. Окрім того, видно ознаки перенавчання НМ за функцій активації Tanh та Softplus.

Таблиця 2

## Дослідження НМ за різних функцій активації на навчальній вибірці зі 140 прикладів

Функція активації		Linear	Sigmoid	Tanh	Softplus	ReLU	L-ReLU
Навчальна вибірка	$\rho_{opt}$	0,28	0,58	1	0,78	1	1
	$\rho_{кор}$	0,32	0,61	1	0,78	1	1
	$H$	0,3	0,6	1	0,78	1	1
Тестова вибірка	$\rho_{opt}$	0,07	0,17	0,03	0,1	0,16	0,16
	$\rho_{кор}$	0,11	0,29	0,05	0,18	0,25	0,28
	$H$	0,09	0,21	0,04	0,13	0,19	0,2

## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

Із таблиці видно, що на малих обсягах даних найбільш успішно на тестовій вибірці показали себе функції активації Sigmoid, ReLU та Leaky-ReLU. Найкращий результат точності на навча-

льних даних дали функції активації Tanh, ReLU, Leaky-ReLU, тому їх можна вважати перспективними для більших обсягів вибірки.

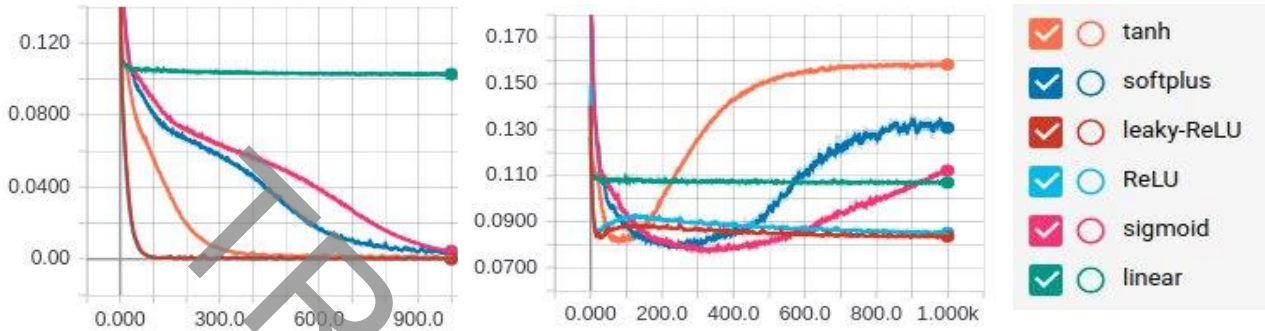


Рис. 12. Графік залежності помилки MSE від епохи для навчальної та тестової вибірок з 1 400 прикладів

Таблиця 3

## Дослідження НМ за різних функцій активації на навчальній вибірці з 1 400 прикладів

Функція активації		Linear	Sigmoid	Tanh	Softplus	ReLU	L-ReLU
Навчальна вибірка	$\rho^{ont}$	0,2	0,98	1	0,99	1	1
	$\rho^{kop}$	0,24	0,99	1	0,99	1	1
	$H$	0,22	0,98	1	0,99	1	1
Тестова вибірка	$\rho^{ont}$	0,19	0,53	0,06	0,3	0,46	0,46
	$\rho^{kop}$	0,24	0,7	0,06	0,35	0,61	0,61
	$H$	0,21	0,6	0,6	0,32	0,53	0,52

Із таблиці видно, що на середніх обсягах даних найбільш успішно на тестовій вибірці показали себе функції активації Sigmoid, Tanh,

ReLU та Leaky-ReLU. Окрім лінійної функції активації, усі функції дали максимальну точність на навчальних даних.

Таблиця 4

## Дослідження НМ за різних функцій активації на навчальній вибірці з 14 000 прикладів

Функція активації		Linear	Sigmoid	Tanh	Softplus	ReLU	L-ReLU
Навчальна вибірка	$\rho^{ont}$	0,2	1	0,99	0,99	1	1
	$\rho^{kop}$	0,24	1	0,99	0,99	1	1
	$H$	0,22	1	0,99	0,99	1	1
Тестова вибірка	$\rho^{ont}$	0,19	0,66	0,57	0,61	0,76	0,74
	$\rho^{kop}$	0,24	0,78	0,63	0,67	0,86	0,83
	$H$	0,21	0,71	0,6	0,64	0,81	0,79

Із таблиці видно, що на обсягах даних більших за середні найбільш успішно на тестовій вибірці показали себе функції активації

Sigmoid, ReLU та Leaky-ReLU. Окрім лінійної функції активації, усі функції показали максимальну точність на навчальних даних.

МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

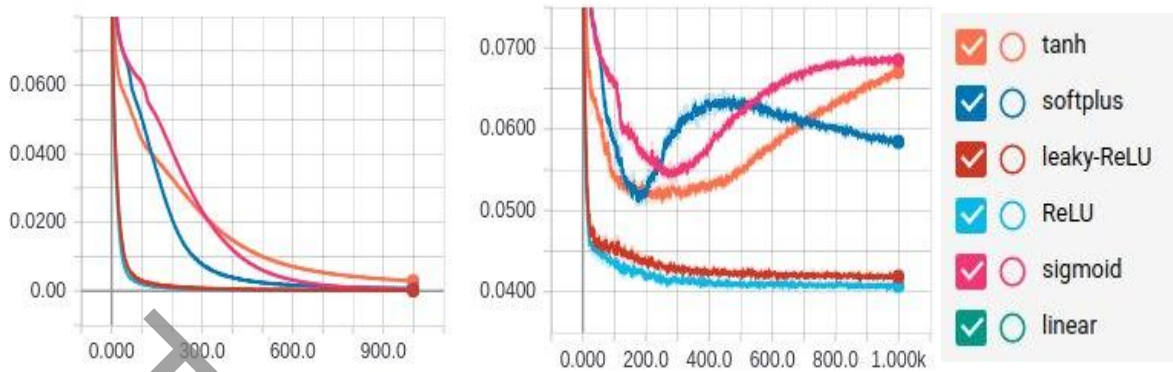


Рис. 13. Графік залежності помилки MSE від епохи для навчальної й тестової вибірок з 14 000 прикладів

Таблиця 5

Дослідження НМ за різних функцій активації на навчальній вибірці зі 49 000 прикладів

Функція активації		Linear	Sigmoid	Tanh	Softplus	ReLU	L-ReLU
Навчальна вибірка	$\rho_{opt}$	0,2	1	0,96	1	1	1
	$\rho_{кор}$	0,24	1	0,96	1	1	1
	$H$	0,22	1	0,96	1	1	1
Тестова вибірка	$\rho_{opt}$	0,19	0,81	0,8	0,76	0,83	0,85
	$\rho_{кор}$	0,24	0,86	0,84	0,8	0,9	0,9
	$H$	0,21	0,83	0,82	0,78	0,86	0,88

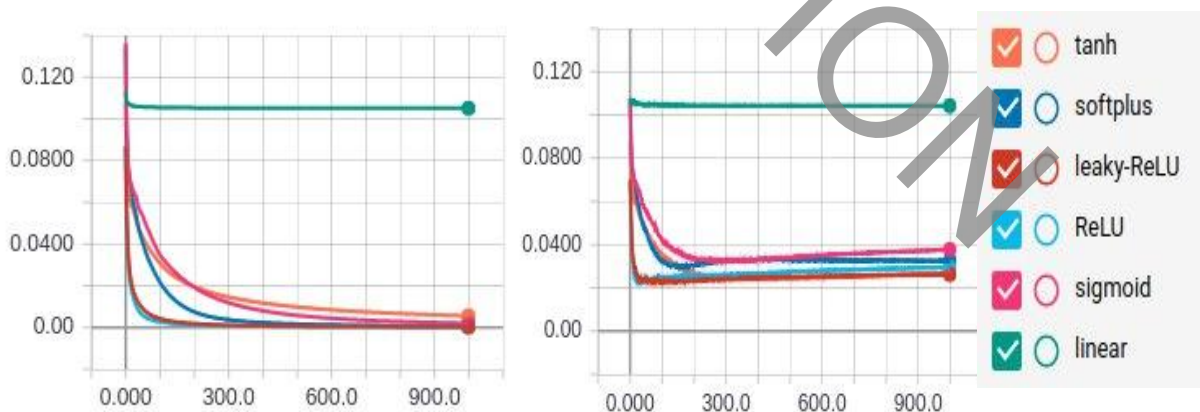


Рис. 14. Графік залежності помилки MSE від епохи для навчальної й тестової вибірок зі 49 000 прикладів

Результуюча НМ конфігурації 34-2-410-34 з функцією активації Leaky-ReLU ( $\alpha = 0,1$ ) у прихованому шарі та лінійною функцією активації у вихідному шарі після навчання на

49 000 прикладів протягом 1 000 епох досягла значення MSE в 0,0024 на контрольній вибірці та у 86 % визначає оптимальний шлях.

## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

**Наукова новизна та практична значимість**

Досліджено НМ конфігурації 34-2-410-34 з функцією активації Leaky-ReLU ( $\alpha = 0,1$ ) у прихованому шарі та лінійною – у вихідному. Експерименти проведені для різних алгоритмів оптимізації навчання протягом 100 епох за обсягу навчальної вибірки зі 49 000 прикладів.

Результати експериментів наведені у табл. 6, процес навчання проілюстровано на рис. 15.

За результатами експериментів видно, що НМ за класичним градієнтним спуском навчається дуже повільно, НМ за стохастичним градієнтним спуском демонструє значне покращення. Алгоритми навчання Adam, AdaMax та Nadam показали майже ідентичні результати.

Таблиця 6

Дослідження НМ конфігурації 34-2-410-34 за різними алгоритмами

Алгоритм		BGD	MB SGD	Adam	Adamax	Nadam
Навчальна вибірка	$\rho^{opt}$	0	0,09	0,99	0,96	0,99
	$\rho^{кор}$	0	0,1	1	0,97	0,99
	$H$	0	0,1	0,99	0,96	0,99
Тестова вибірка	$\rho^{opt}$	0	0,11	0,87	0,85	0,87
	$\rho^{кор}$	0	0,12	0,92	0,9	0,92
	$H$	0	0,11	0,89	0,87	0,9

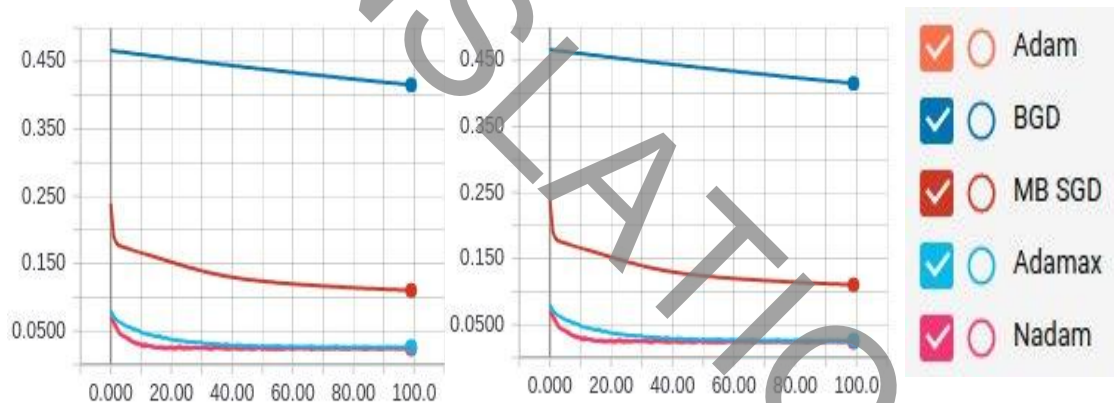


Рис. 15. Графік залежності помилки MSE від епохи для навчальної й тестової вибірок для різних алгоритмів навчання

**Висновки**

1. Розв'язана задача маршрутизації в об'єднаній комп'ютерній мережі на основі розробленої програмної моделі «MLP 34-2-410-34» з використанням мови Python та фреймворку TensorFlow, що дозволяє генерувати вибірки: навчальну (140, 1 400, 14 000, 49 000 прикладів); тестову (700 прикладів); контрольну (700 прикладів), а також моделювати роботу нейронної мережі та провести дослідження її параметрів.

2. Досліджена ефективність на основі середнього гармонійного НМ конфігурації 34-2-X-34 із сигмоїдною функцією активації у прихованому шарі та лінійною – у вихідному за різної кількості прихованих нейронів: 62; 410; 760; 1 110; 1 456. Під час навчання НМ використаний сучасний алгоритм оптимізації стохастичного градієнтного спуску Adam із рекомендованими гіперпараметрами: швидкість навчання  $\alpha = 0,001$ ; інерція  $\beta_1 = 0,9$ ; RMSprop  $\beta_2 = 0,999$ ;  $\text{decau} = 10^{-5}$ . Дослідження показали, що точність нейронної мережі може бути

## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

досягнена за 410 прихованих нейронів, а подальше збільшення майже ніяк не поліпшує результати.

3. Досліджено ефективність на основі середнього гармонійного НМ конфігурації 34-2-410-34 за різними функціями активації: лінійною; сигмоїдною; гіперболічним тангенсом; Softplus, ReLU; Leaky-ReLU із використанням алгоритму Adam на навчальних вибірках різного обсягу (140, 1 400, 14 000, 49 000 прикладів). Дослідження показало, що функції активації ReLU та Leaky-ReLU навчаються найбільш швидко на всіх обсягах навчальної вибірки та менше за інші функції активації піддаються перенавчанню. НМ конфігурації 34-2-410-34 за функцій активації Tanh та Softplus здатна досягнути 100 % точності на навчальній вибірці, але ці функції навчаються повільніше за ReLU та

Leaky-ReLU й дуже піддаються перенавчанню за незначних обсягів навчальної вибірки (140, 1 400 та 14 000 прикладів). Під час використання сигмоїдної функції активації нейронна мережа також перенавчається, але за значного обсягу навчальних прикладів (49 000 прикладів) вона здатна досягнути точності (83 %), близької до Leaky-ReLU (88 %) або ReLU (86 %).

4. Досліджена ефективність НМ конфігурації 34-2-410-34 з функцією активації Leaky-ReLU ( $\alpha = 0,1$ ) у прихованому шарі та лінійною – у вихідному. Експерименти проведені за різними алгоритмами оптимізації (BGD, MB SGD, Adam, Adamax, Nadam) протягом 100 епох за обсягу навчальної вибірки у 49 000 прикладів. Алгоритми Adam, Adamax та Nadam показали майже ідентичні результати, точність складає 89, 87 та 90 % відповідно.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Колесніков, К. В. Аналіз результатів дослідження реалізації задачі маршрутизації на основі нейронних мереж та генетичних алгоритмів / К. В. Колесніков, А. Р. Карапетян, В. Ю. Баган // Вісн. Черкас. держ. технол. ун-ту. Серія: Технічні науки. – 2016. – № 1. – С. 28–34.
2. Кутыркин, А. В. Использование нейронной сети Хопфилда для решения оптимизационных задач маршрутизации : метод. указания / А. В. Кутыркин, А. В. Семин. – Москва : Изд-во Моск. гос. ун-та путей сообщения, 2007. – 15 с.
3. Павленко, М. А. Анализ возможностей искусственных нейронных сетей для решения задач однопутевой маршрутизации в ТКС [Електронний ресурс] // Проблеми телекомунікацій. – 2011. – № 2 (4). – Режим доступу: <http://pt.journal.kh.ua/index/0-139> – Назва з екрана. – Перевірено : 20.11.2018.
4. Палмер, М. Проектирование и внедрение компьютерных сетей / М. Палмер, Р. Б. Синклер. – Санкт-Петербург : БХВ-Петербург, 2004. – 752 с.
5. Пахомова, В. М. Аналіз методів з природними механізмами визначення оптимального маршруту в комп'ютерній мережі Придніпровської залізниці / В. М. Пахомова, Р. О. Лепеха // Інформ.-керуючі системи на залізн. трансп. – 2014. – № 4. – С. 82–91.
6. Пахомова, В. М. Рішення задачі маршрутизації в комп'ютерній мережі Придніпровської залізниці на основі нейронної моделі Хопфільда / В. М. Пахомова, Ю. О. Федоренко // Інформ.-керуючі системи на залізн. трансп. – 2012. – № 4. – С. 76–84.
7. Реалізація задачі вибору оптимального маршруту нейронною мережею Хопфільда / А. М. Бриндас, П. І. Рожак, Н. О. Семинішин, Р. Р. Курка // Наук. вісн. НЛТУ України : зб. наук.-техн. пр. – Львів, 2016. – Вип. 26.1. – С. 357–363.
8. Хайкин, С. Нейронные сети. Полный курс : [пер. с англ.] / С. Хайкин. – 2-е изд., испр. – Москва : Вильямс, 2006. – 1104 с.
9. An open source machine learning framework for everyone [Електронний ресурс] : [веб-сайт] / TensorFlow. – Електрон. текст. дані. – Режим доступу: <https://www.tensorflow.org> – Назва з екрана. – Перевірено : 05.06.2018.
10. A Survey of Artificial Immune System Based Intrusion Detection / Hua Yang, Tao Li, Xinlei Hu, Feng Wang, Yang Zou // The Scientific World Journal. – 2014. – Vol. 2014. – P. 1–11. doi: 10.1155/2014/156790
11. CiscoTips [Електронний ресурс] : [веб-сайт]. – Електрон. текст. дані. – Режим доступу: <http://ciscotips.ru/ospf> – Назва з екрана. – Перевірено : 20.05.2018.

## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

12. Hopfield, J. J. Neural networks and physical systems with emergent collective computational abilities / John J. Hopfield // *Proceedings of National Academy of Sciences*. – 1982. – Vol. 79. – Iss. 8. – P. 2554–2558. doi: 10.1073/pnas.79.8.2554
13. IBM [Електронний ресурс] : [веб-сайт] / IBM Knowledge Center. – Електрон. текст. дані. – Режим доступу: <https://u.to/G-giFA> – Назва з екрана. – Перевірено : 20.05.2018.
14. Keras [Електронний ресурс] : [веб-сайт]. – Електрон. текст. дані. – Режим доступу: <https://keras.io> – Назва з екрана. – Перевірено : 05.06.2018.
15. Neural Network Based Near-Optimal Routing Algorithm / Chang Wook Ahn, R. S. Ramakrishna, In Chan Choi, Chung Gu Kang // *Neural Information Processing, 2002. ICONIP'02 : Proc. of the 9th Intern. Conf. (18–22 Nov. 2002)*. – Singapore, 2002. – Vol. 5. – P. 1771–1776. doi: 10.1109/iconip.2002.1198978
16. New algorithm for packet routing in mobile ad-hoc networks / N. S. Kojić, M. B. Zajeganović-Ivančić, I. S. Reljin, B. D. Reljin // *Journal of Automatic Control*. – 2010. – Vol. 20. – Iss. 1. – P. 9–16. doi: 10.2298/jac100109k
17. Pakhomova, V. M. Network Traffic Forecasting in information-telecommunication System of Prydniprovsk Railways Based on Neuro-fuzzy Network // *Наука та прогрес транспорту*. – 2016. – № 6 (66). – С. 105–114. doi: 10.15802/stp2016/90485
18. Schuler, W. H. A novel hybrid training method for hopfield neural networks applied to routing in communications networks / W. H. Schuler, C. J. A. Bastos-Filho, A. L. I. Oliveira // *International Journal of Hybrid Intelligent Systems*. – 2009. – Vol. 6. – Iss. 1. – P. 27–39. doi: 10.3233/his-2009-0074
19. Security Lab.ru [Електронний ресурс] : [веб-сайт]. – Електрон. текст. дані. – Режим доступу: <https://www.securitylab.ru> – Назва з екрана. – Перевірено : 20.05.2018.
20. Zhukovyts'kyu, I. Research of Token Ring network options in automation system of marshalling yard / I. Zhukovyts'kyu, V. Pakhomova // *Transport Problems*. – 2018. – Vol. 13. – Iss. 2. – P. 145–154.

В. Н. ПАХОМОВА<sup>1\*</sup>, И. Д. ЦЫКАЛО<sup>2\*</sup>

<sup>1\*</sup>Каф. «Електронные вычислительные машины», Днепропетровский национальный университет железнодорожного транспорта имени академика В. Лазаряна, ул. Лазаряна, 2, Днипро, Украина, 49010, тел. +38 (056) 373 15 89, эл. почта [viknikpakh@gmail.com](mailto:viknikpakh@gmail.com), ORCID 0000-0002-0022-099X

<sup>2\*</sup>Каф. «Електронные вычислительные машины», Днепропетровский национальный университет железнодорожного транспорта имени академика В. Лазаряна, ул. Лазаряна, 2, Днипро, Украина, 49010, тел. +38 (056) 373 15 89, эл. почта [igor.tsykalo@gmail.com](mailto:igor.tsykalo@gmail.com), ORCID 0000-0002-1629-5873

## ОПРЕДЕЛЕНИЕ ОПТИМАЛЬНОГО МАРШРУТА В КОМПЬЮТЕРНОЙ СЕТИ СРЕДСТВАМИ МНОГОСЛОЙНОЙ НЕЙРОННОЙ МОДЕЛИ

**Цель.** Классические алгоритмы поиска кратчайшего пути на графе, что лежат в основе существующих протоколов маршрутизации, которые сегодня используют в компьютерных сетях, в условиях постоянного изменения загруженности сети не могут привести к оптимальному решению в реальном времени. В связи с этим целью статьи является разработать методику определения оптимального маршрута в объединенной компьютерной сети. **Методика.** Для определения оптимального маршрута в объединенной компьютерной сети, которая работает по разным технологиям, написана на языке Python с использованием фреймворка TensorFlow программная модель «MLP 34-2-410-34». Она позволяет выполнять следующие этапы: генерацию выборки (случайную или сбалансированную); создание нейронной сети, на вход которой подаются массив пропускных способностей каналов компьютерной сети; обучение и тестирование нейронной сети на соответствующих выборках. **Результаты.** Нейронная сеть конфигурации 34-2-410-34 с функциями активации ReLU и Leaky-ReLU в скрытом слое и линейной функцией активации в результирующем слое обучается по алгоритму Adam. Этот алгоритм является комбинацией алгоритмов Adagrad, RMSprop и стохастического градиентного спуска с инерцией. Указанные функции учатся наиболее быстро на всех объемах учебной выборки, меньше других поддаются переобучению, и достигают значения ошибки в 0,0024 на контрольной выборке и в 86 % определяют оптимальный путь. **Научная новизна.** Проведено исследование параметров нейронной сети на основе расчета среднего гармоничного при разных функциях активации (Linear, Sigmoid,

Creative Commons Attribution 4.0 International

doi: 10.15802/stp2018/154443

© В. М. Пахомова, И. Д. Цыкало, 2018

## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

Tanh, Softplus, ReLU, L-ReLU) на учебных выборках разного объема (140, 1 400, 14 000, 49 000 примеров) и за различными алгоритмами оптимизации обучения нейронной сети (BGD, MB SGD, Adam, Adamax, Nadam). **Практическая значимость.** Использование нейронной модели, на вход которой подаются значения пропускных способностей каналов, позволит в реальном масштабе времени определить оптимальный маршрут в объединенной компьютерной сети.

**Ключевые слова:** компьютерная сеть; оптимальный маршрут; нейронная сеть; выборка; среднее гармоничное; функция активации; алгоритм оптимизации

V. M. PAKHOMOVA<sup>1\*</sup>, I. D. TSYKALO<sup>2\*</sup>

<sup>1\*</sup>Dep. «Electronic Computing Machines», Dnipropetrovsk National University of Railway Transport named after Academician V. Lazaryan, Lazaryan St., 2, 49010, Dnipro, Ukraine, tel. +38 (056) 373 15 89, e-mail viknikpakh@gmail.com, ORCID 0000-0002-0022-099X

<sup>2\*</sup> Dep. «Electronic Computing Machines», Dnipropetrovsk National University of Railway Transport named after Academician V. Lazaryan, Lazaryan St., 2, 49010, Dnipro, Ukraine, tel. +38 (056) 373 15 89, e-mail ihor.tsykalo@gmail.com, ORCID 0000-0002-1629-5873

## OPTIMAL ROUTE DEFINITION IN THE NETWORK BASED ON THE MULTILAYER NEURAL MODEL

**Purpose.** The classic algorithms for finding the shortest path on the graph that underlie existing routing protocols, which are now used in computer networks, in conditions of constant change in network traffic cannot lead to the optimal solution in real time. In this regard, the purpose of the article is to develop a methodology for determining the optimal route in the unified computer network. **Methodology.** To determine the optimal route in the computer network, the program model "MLP 34-2-410-34" was developed in Python using the TensorFlow framework. It allows to perform the following steps: sample generation (random or balanced); creation of a neural network, the input of which is an array of bandwidth of the computer network channels; training and testing of the neural network in the appropriate samples. **Findings.** Neural network of 34-2-410-34 configuration with ReLU and Leaky-ReLU activation functions in a hidden layer and the linear activation function in the output layer learns from Adam algorithm. This algorithm is a combination of Adagrad, RMSprop algorithms and stochastic gradient descent with inertia. These functions learn the most quickly in all volumes of the train sample, less than others are subject to re-evaluation, and reach the value of the error of 0.0024 on the control sample and in 86% determine the optimal path. **Originality.** We conducted the study of the neural network parameters based on the calculation of the harmonic mean with different activation functions (Linear, Sigmoid, Tanh, Softplus, ReLU, L-ReLU) on train samples of different volumes (140, 1400, 14000, 49000 examples) and with various neural network training algorithms (BGD, MB SGD, Adam, Adamax, Nadam). **Practical value.** The use of a neural model, the input of which is an array of channel bandwidth, will allow in real time to determine the optimal route in the computer network.

**Keywords:** computer network; optimal route; neural network; sampling; harmonic mean; activation function optimization algorithm

### REFERENCES

1. Kolesnikov, K. V., Karapetian, A. R., & Bahan, V. Y. (2016). Analiz rezultativ doslidzhennia realizatsii zadachi marshrutyzatsii na osnovi neironnykh merezh ta henetychnykh alhorytmiv. *Visnyk Cherkaskoho derzhavnoho tekhnolohichnoho universytetu. Seriya: Tekhnichni nauky, 1*, 28-34. (in Ukrainian)
2. Kutyркин, A. V., & Semin, A. V. (2007). *Ispolzovanie neyronnoy seti Khopfilda dlya resheniya optimizatsionnykh zadach marshrutyzatsii: Metodicheskie ukazaniya*. Moscow: Izdatelstvo Moskovskogo gosudarstvennogo universiteta putey soobshcheniya. (in Russian)
3. Pavlenko M. A. (2011). Analysis opportunities of artificial neural networks for solving single-path routing in telecommunication network. *Problemy telekomunikatsii, 2(4)*. Retrived from <http://pt.journal.kh.ua/index/0-139> (in Russian)
4. Palmer, M., & Sinkler, R. B. (2004). *Proektirovanie i vnedrenie kompyuternykh setey*. St. Petersburg: BKhV-Peterburg. (in Russian)



## МОДЕЛЮВАННЯ ЗАДАЧ ТРАНСПОРТУ ТА ЕКОНОМІКИ

5. Pakhomova, V. M., & Lepekha, R. O. (2014). Analiz metodiv z pryrodnymy mekhanizmy vyznachennia optymalnoho marshrutu v komp'uternii mrezhi Prydniprovskoi zaliznytsi. *Information and control systems at railway transport*, 4, 82-91. (in Ukrainian)
6. Pakhomova, V. M., & Fedorenko, Y. O. (2012). Rishennia zadachi marshrutzatsii v komp'uternii mrezhi Prydniprovskoi zaliznytsi na osnovi neironnoi modeli Khopfilda. *Information and control systems at railway transport*, 4, 76-84. (in Ukrainian)
7. Bryndas, A. M., Rozhak, P. I., Semynshyn, N. O., & Kurka, R. R. (2016). Realizatsiia zadachi vyboru optymalnoho marshrutu neironnoi mrezheiu Khopfilda. *Naukovyi visnyk NLTU Ukrainy*, 26(1), 357-363. (in Ukrainian)
8. Khaykin, S. (2006). *Neyronnye seti. Polnyy kurs*. Moscow: Vilyams. (in Russian)
9. An open source machine learning framework for everyone. *TensorFlow*. Retrieved from <https://www.tensorflow.org> (in English)
10. Yang, H., Li, T., Hu, X., Wang, F., & Zou, Y. (2014). A Survey of Artificial Immune System Based Intrusion Detection. *The Scientific World Journal*, 2014, 1-11. doi: 10.1155/2014/156790 (in English)
11. CiscoTips. Retrieved from <http://ciscotips.ru/ospf> (in Russian)
12. Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8), 2554-2558. doi: 10.1073/pnas.79.8.2554 (in English)
13. IBM. *IBM Knowledge Center*. Retrieved from <https://u.to/G-giFA> (in Russian)
14. Keras. Retrieved from <https://keras.io> (in English)
15. Chang Wook Ahn, Ramakrishna, R. S., In Chan Choi, & Chung Gu Kang. (n.d.). Neural network based near-optimal routing algorithm. *Proceedings of the 9th International Conference on Neural Information Processing, 2002. ICONIP'02*. doi: 10.1109/iconip.2002.1198978 (in English)
16. Kojic, N., Zajeganovic-Ivancic, M., Reljin, I., & Reljin, B. (2010). New algorithm for packet routing in mobile ad-hoc networks. *Journal of Automatic Control*, 20(1), 9-16. doi: 10.2298/jac1001009k (in English)
17. Pakhomova, V. M. (2016). Network Traffic Forecasting in Information-telecommunication System of Prydniprovsk Railways Based on Neuro-fuzzy Network. *Science and Transport Progress*, 6(66), 105-114. doi: 10.15802/stp2016/90485 (in English)
18. Schuler, W. H., Bastos-Filho, C. J. A., & Oliveira, A. L. I. (2009). A novel hybrid training method for hopfield neural networks applied to routing in communications networks. *International Journal of Hybrid Intelligent Systems*, 6(1), 27-39. doi: 10.3233/his-2009-0074 (in English)
19. Security Lab.ru. Retrieved from <https://www.securitylab.ru> (in Russian)
20. Zhukovyts'kyu, I., & Pakhomova, V. (2018). Research of Token Ring network options in automation system of marshalling yard. *Transport Problems*, 13(2), 145-154. (in English)

Надійшла до редколегії: 27.07.2018

Прийнята до друку: 06.11.2018